QUANTUM GATE FOR CARRYING OUT A GROVER'S QUANTUM ALGORITHM AND A RELATIVE METHOD OF PERFORMING THE INTERFERENCE OPERATION OF A GROVER'S QUANTUM ALGORITHM

Cross-Reference to Related Applications

[0001] The present application is a continuation-in-part of U.S. Patent Application Serial No. TBD filed on November 4, 2003 which in turn is a continuation-in-part of U.S. Patent Application Serial No. 10/615,446 filed July 8, 2003, the entire contents of which are incorporated herein by reference.

Field of the Invention

[0002] The present invention relates to quantum algorithms, and more precisely, to a quantum gate for carrying out a Grover's quantum algorithm with any number n of qubits and a relative method for carrying out the interference operation of the Grover's quantum algorithm.

Background of the Invention

[0003] Quantum algorithms are global random searching algorithms based on the principles, laws and effects of quantum mechanics. They are used for controlling a process or for processing data in a database, and more specifically, for controlling a process that may include search-of-minima intelligent

operations.

[0004] In the quantum search, each design variable is represented by a finite linear superposition of initial states, with a sequence of elementary unitary steps manipulating the initial quantum state $|i\rangle$ (for the input) such that a measurement of the final state of the system yields the correct output. Usually, three principle operators, i.e., linear superposition (coherent states), entanglement and interference are used in the quantum search algorithm.

[0005] A general description of quantum algorithms is contained in U.S. patent application number 10/615,446 and in the European patent application no. 02425672.9. Both of these applications are assigned to the current assignee of the present invention.

[0006] For a better understanding, a detailed description of the Grover's quantum algorithm is presented below. Grover's problem may be stated as follows:

Input	A function $f:\{0,1\}^n \to \{0,1\}$ such that $\exists x \in \{0,1\}^n:$ $(f(x)=1 \wedge \forall y \in \{0,1\}^n : x \neq y \Rightarrow f(y)=0)$
Problem	Find x

[0007] In a Deutsch-Jozsa's algorithm there are two classes of input functions and it must be determined what class the input function belongs to. In this case the problem is in some sense identical in its form, even if it is more difficult because now we are dealing with 2^n classes of input functions (each function of the kind described forms a class).

[0008] The diagram of the Grover's algorithm is

depicted in FIG. 1, and the gate equation is

$$\Phi = \left[(D_n \otimes I) \cdot U_F \right]^h \cdot \left({}^{n+1}H \right) \qquad (1)$$

Operator D_n is called a diffusion matrix of order n and is responsible for interference in this algorithm. The diffusion matrix is defined as follows:

D_n	|0..0>	|0..1>	...	|i>	...	|1..0>	|1..1>
|0..0>	$-1+1/2^{n-1}$	$1/2^{n-1}$	...	$1/2^{n-1}$	...	$1/2^{n-1}$	$1/2^{n-1}$
|0..1>	$1/2^{n-1}$	$-1+1/2^{n-1}$	...	$1/2^{n-1}$	...	$1/2^{n-1}$	$1/2^{n-1}$
...	...	...	...	...	...	...	...
|i>	$1/2^{n-1}$	$1/2^{n-1}$	...	$-1+1/2^{n-1}$	...	$1/2^{n-1}$	$1/2^{n-1}$
...	...	...	...	...	...	...	...
|1..0>	$1/2^{n-1}$	$1/2^{n-1}$	...	$1/2^{n-1}$	...	$-1+1/2^{n-1}$	$1/2^{n-1}$
|1..1>	$1/2^{n-1}$	$1/2^{n-1}$	...	$1/2^{n-1}$	...	$1/2^{n-1}$	$-1+1/2^{n-1}$

[0009] Grover's algorithm may be implemented in routines for searching a desired item in a set, by representing in vector form each item of the set forming an input set of vectors, and applying a Grover's algorithm to this set of vectors. The output vector represents the desired item.

[0010] The implementation of the Grover's algorithm clearly implies the calculation of several vector products. In fact, all qubits must be multiplied by the Hadamard matrix H, then by the entanglement matrix U_F, and all qubits but the latter must be multiplied by matrix D_n.

[0011] These multiplications could be carried out via software, but it is quite evident that the number of qubits of a quantum algorithm is very critical in terms of computational speed. In fact, referring to the scheme in FIG. 1, it must be noted that the addition of only one qubit doubles the dimensions of matrices.

Thus, the number of elements (and products) increases exponentially.

[0012] A method of performing the superposition operation of a Grover's or Deutsch-Jozsa's quantum algorithm over an input set of vectors is disclosed in the European patent application no. 01830383.4, which is assigned to the current assignee of the present invention. This method exploits the fact that any rotated vector obtained performing the Hadamard rotation H (on an input vector) contemplated by the superposition operation of these quantum algorithms can be easily encoded in a binary vector. Therefore, the successive tensor product of the rotated vectors for generating linear superposition vectors can be carried out by logic gates. This fact allows a noticeable time saving because logic gates are very fast.

[0013] However, this is not sufficient to significantly speed up running these quantum algorithms because the entanglement matrix U_F is a $2^{n+1} \times 2^{n+1}$ square matrix, which implies a considerable computational weight both in the Grover's algorithm as well as in the Deutsch-Jozsa's algorithm.

[0014] Differently from other quantum algorithms, in the Grover's algorithm it is possible to iterate h times the entanglement and interference operations until the best solution is reached. An example of evolution of the Grover's algorithm with n = 3 is given in FIG. 2a, in which basis vector (Step 0) and superposition (Step 1), entanglement (Step 2) and interference (Step 3) output vectors are reported in order. Iterating the entanglement and interference operations produces a better distribution of probability amplitudes.

[0015] Each value is a component on the output vector referred to a basis of vectors of n+1 qubits. There are couples or pairs of values of opposite sign, referred to vectors of the basis having in common the first (leftmost) n qubits. For example, the values 0.625 and -0.625 are referred to vectors $|0110\rangle$ and $|0111\rangle$, respectively. Each pair of elements having an opposite sign represents the probability amplitude of a certain element of the database. For the considered example, the value 0.625 is the probability of the element associated to vector $|011\rangle$ after 3 iterations (h = 3).

[0016] The algorithm may be iterated as far as a certain quantity is to be minimized, calculated as a function of the components of the output vector, and is smaller than a certain pre-established value. For instance, this quantity can be the Shannon entropy:

$$S(h) = -\sum_{k=1}^{2^{n+1}} \|q_k(h)\|^2 \log\|q_k(h)\|^2 \qquad (2)$$

where $q_k(h)$ is the k-th component of the output vector Q taken after h iterations.

[0017] The components of the output vector obtained after h = 15 iterations are represented in FIG. 2b. From FIG. 2b it is clear that the element of the database to be found is associated to vector $|011\rangle$, and after 15 iterations the Grover's quantum algorithm will find it with a probability of about 0.69.

[0018] From the above discussion it is evident that the large number of computations for the Grover's

algorithm represents a considerable burden because multiplications by the entanglement matrix U_F and the interference matrix $D_n \otimes I$ might be repeated many (h) times to produce the best result, and the process may take considerable time.

Summary of the Invention

[0019] In view of the foregoing background, an object of the present invention is to provide a quantum gate and a relative method for executing Grover's quantum algorithms in a very fast manner.

[0020] This and other objects, advantages and features in accordance with the present invention are provided by a quantum gate for carrying out a Grover's quantum algorithm using a certain binary function (f) defined on a space having a vector basis of n qubits. The quantum gate may comprise a superposition subsystem for carrying out a superposition operation on components of input vectors for generating components of superposition vectors on a second vector basis of $n+1$ qubits. An entanglement subsystem may carry out an entanglement operation on components of the linear superposition vectors for generating components of entanglement vectors. An interference subsystem may carry out an interference operation on components of the entanglement vectors for generating components of output vectors.

[0021] The quantum gate is capable of performing the interference operation of the Grover's algorithm in a very fast manner by using an adder being input with signals representing even or odd components of an entanglement vector, and generating a sum signal representing a weighted sum with a scale factor of the

even or odd components. The quantum gate also uses an array of adders, each being input with a respective signal representative of an even or odd component of an entanglement vector, and with the weighted sum signal, and generating a signal representative of an even or odd component of an output vector as the difference between the weighted sum signal and the signal representing an even or odd component of an entanglement vector.

[0022] Another aspect of the present invention is directed to a method for carrying out an interference operation of a Grover's quantum algorithm comprising the steps of calculating a weighted sum with a certain scale factor of even or odd components of an entanglement vector, and generating each even or odd component of an output vector by subtracting from the weighted sum corresponding even or odd components of an entanglement vector.

Brief Description of the Drawings

[0023] The particular aspects and advantages of the present invention will become more evident through the following description of several important embodiments and by referring to the attached drawings, wherein:

[0024] FIG. 1 is an example of a circuit forming a Grover's quantum gate in accordance with the prior art;

[0025] FIGS. 2a and 2b illustrate the evolution of a Grover's quantum algorithm in accordance with the prior art;

[0026] FIG. 3 is a detailed view of entanglement and interference subsystems of the quantum gate in accordance with the present invention;

[0027] FIG. 4 is an embodiment of the adder HB25 for

the even or odd components of an entanglement vector for $n = 3$ of section I-b of FIG. 3;

[0028] FIG. 5 is a circuit used for forming the adder HB25 of section I-a of FIG. 3 according to a preferred modular embodiment of the quantum gate;

[0029] FIG. 6 is a preferred embodiment of the adder HB25 of section I-a of FIG. 3 according to a modular architecture for $n = 4$;

[0030] FIG. 7 is a preferred embodiment of a single element of section I-c of FIG. 3;

[0031] FIG. 8 is a view of a module and the microprocessor unit of the quantum gate in accordance with the present invention;

[0032] FIG. 9 illustrates the digital circuit LOGIC of FIG. 8;

[0033] FIG. 10 shows the array architecture of the inverting circuit INVERT of FIG. 8; and

[0034] FIG. 11 is a preferred embodiment of the inverters of FIG. 8.

Detailed Description of the Preferred Embodiments

[0035] A quantum gate for fast running quantum algorithms applied over a set of input vectors is disclosed in the European patent application no. 02425672.9 which is assigned to the current assignee of the present invention. The quantum gate is comprises a superposition subsystem carrying out a linear superposition, an entanglement subsystem carrying out an entanglement operation and an interference subsystem carrying out an interference operation according to the Grover's quantum algorithm.

[0036] A hardware quantum gate of the present invention for performing Grover's algorithm with any

number of iterations comprises two parts. Part I: (analog) for performing a calculation step-by-step of output values. This first part is divided in the following subsections: I-a: Entanglement; and I-b and I-c: Interference. Part II: (digital) for performing entropy evaluation, storage of vectors for iterations and output display. This part also provides the first basis of vectors.

[0037] An analog part for a three-qubits quantum gate is depicted in FIG. 3. A command circuit HB14 generates eight command signals Vc1, ..., Vc8 each representing a value of the function *f(.)* to be processed on a respective vector of the first basis.

[0038] The entanglement subsystem, which preferably may be formed by the command circuit HB14 and by an array of multiplexers I-a, is input with the voltage signals O11, ..., O82 representing the sixteen components of a linear superposition vector, and generates the signals Q1, ..., Q8 representing only the even or the odd components of an entanglement vector.

[0039] Let us assume that these signals represent the odd components of an entanglement vector. It is not necessary to calculate all components of the entanglement or output vectors since the odd components of any vector are always opposite to the even components. Therefore, entanglement and interference operations are carried out only on the odd components. The other components are calculated by simply inverting the first ones.

[0040] The sections I-b and I-c of a quantum gate shown in FIG. 3 allow the interference operation of the Grover's quantum algorithm to be carried out very quickly. The matrix $D_n \otimes I$ has the following properties.

Odd columns (or rows, because $D_n \otimes I$ is symmetric) have non-zero odd components and even columns have non-zero even components. The value of all non-zero components, except for the i^{th} component of i^{th} column (diagonal elements), is $1/2^{n-1}$. The components on the up-left down-right diagonal of the matrix differ form the other non-zero components by being decreased by 1. G^* is an entanglement vector. The output vector of the quantum algorithm $V = (D_n \otimes I) G^*$ involves only a weighted sum of components of G^*. The value $1/2^{n-1}$ depends only from the number n of qubits.

[0041] From the above analysis, the generic element v_i of V can be written as follows as a function of components g_i^* of the entanglement vector G^*:

$$v_i = \begin{cases} \frac{1}{2^{n-1}} \sum_{j=1}^{2^n} g_{2j-1}^* - g_i^* & \text{for } i \text{ odd} \\ \frac{1}{2^{n-1}} \sum_{j=1}^{2^n} g_{2j}^* - g_i^* & \text{for } i \text{ even} \end{cases} \tag{3}$$

Therefore, to calculate a component v_i of the output vector, according to the method of the invention, it is sufficient to calculate a weighted sum of even ($\frac{1}{2^{n-1}} \sum_{j=1}^{2^n} g_{2j}^*$) or odd ($\frac{1}{2^{n-1}} \sum_{j=1}^{2^n} g_{2j-1}^*$) components of the entanglement vector and to subtract from the weighted sum the corresponding component g_i^* of the entanglement vector.

[0042] An adder HB25, which may be as detailed in FIG. 4, sums these components with a certain scale factor that depends only on the number n of qubits (which is 0.25 for $n = 3$), and generates a signal SQ

representing the sum of the odd (or even) components of the entanglement vector. The reference voltage of this adder is 2.5V. This makes the voltage signal representing the scaled sum SQ range between [0÷5] Volts, which is the voltage range of the signals in the digital Part II of the quantum gate.

[0043] The hardware structure of such a quantum gate must be designed for a pre-established number n of qubits, and is used for handling vectors having a different number of qubits. In fact, the adder of FIG. 4 is specifically designed for a certain number n of qubits (in the considered example $n = 3$).

[0044] This lack of flexibility is overcome by forming a quantum gate comprising a plurality of interconnected modules. Each module comprises an adder HB25 having a voltage buffer with an operational amplifier, and the amplifiers of the voltage buffers of all modules are connected in parallel. Basically, each voltage buffer comprises an operational amplifier, a resistor coupling an input of the amplifier to a node at a voltage to be summed and scaled, and a feedback resistor connected between an input and an output of the amplifier.

[0045] A sample embodiment of such a voltage buffer is shown in FIG. 5. The resistors may have different values from that of FIG. 5, provided that the circuit operates as a stand-alone buffer and forms a part of a modular adder when the terminals of the amplifier are connected with corresponding terminals of other similar buffers.

[0046] Preferably, the resistors have a relatively high resistance to prevent the parallel resistors of all the buffers from having an excessively small value

such that the operational amplifier will not function correctly. As it will be evident to those skilled in the art, different architectures of the voltage buffer are possible, provided that they have an operational amplifier, a feedback resistor and another resistor for coupling an input of the amplifier to a node at a voltage to be summed.

[0047] Preferably, the adder HB25 of each module is composed of an auxiliary adder, as that of FIG. 4, for generating a partial weighted sum SQMX with a certain scale factor $1/2^{m-1}$ of a certain number of components 2^m of even or odd components of an entanglement vector, and a voltage buffer, as that depicted in FIG. 5, being input with the voltage signal representative of the weighted sum SQMX. This configuration is particularly convenient because it allows a weighted sum SQ of 2^{m+k} components to be generated by connecting 2^k modules in parallel.

[0048] FIG. 6 shows how two adders HB25 for $n = 3$ qubits may be connected to form a section I-b suitable for the case of $n = 4$ qubits. Finally, an array of adders I-c (see FIG. 3) generates the signals A1, ..., A8 representative of the odd (or even) components of an output vector by subtracting the components Q1, ..., Q8 from the scaled sum SQ. For example, the adders of section I-c may be formed for example as depicted in FIG. 7.

[0049] A basic scheme of a module of a quantum gate in accordance with the present invention is depicted in the dashed rectangle of FIG. 8. Each module substantially comprises an adder HB25 for generating the signal SQ that can be coupled with the adders of the other modules by connecting in common the pins W, U

and S. An array of adders HB16, ..., HB23 generates components of an interference vector A. Each module further comprises a digital circuit LOGIC and an inverting circuit INVERT.

[0050] Preferably, each module comprises also a subgroup HB13 of the above mentioned array of multiplexers I-a of the entanglement subsystem. The logic command signals Vc1, ..., Vc8 encoding the values of the binary function $f(.)$ to be processed with the Grover's algorithm are sent to each subgroup of multiplexers HB13 of the modules through the internal bus BUS.

[0051] The digital circuit LOGIC, belonging to the digital Part II of the quantum gate, generates odd (or even) components of an output vector IN from the respective components of the interference vector A and communicates through the internal bus BUS with a microprocessor unit CPLD.

[0052] To better understand the functioning of the digital circuit LOGIC, reference is made to the more detailed diagram of FIG. 9. The circuit LOGIC comprises an identification circuit SELECTOR that generates a relative identification bit string R and a comparator COMP that compares the strings M and R. When the two strings are equal, the comparator switches active the flag ENABLE, indicating that the microprocessor unit CPLD has selected the module for exchanging data with it through the internal bus BUS.

[0053] The microprocessor unit CPLD performs different types of operations (storing values, evaluating entropy and stopping iteration) in executing the Grover's algorithm. For evaluating the entropy S of the interference vector A, the microprocessor unit CPLD

switches active a first logic signal OUT_EN, thus allowing the A/D converter of the selected module to generate a digital string DA representative of the components of output vector A, calculated by the selected module.

[0054] An analog/digital converter, which for example may be the commercial device ADC0808 of National Semiconductor, receives signals representing the components A1, ..., A8 of the output vector A and produces a corresponding binary string DA. The microprocessor unit may be for example the commercial device XC95288XL of Xilinx, and receives this string and calculates the Shannon entropy. If the Shannon entropy is greater than a pre-established value S, the microprocessor CLPD commands a new iteration of the Grover's algorithm by providing a second logic signal WR that makes the D/A converter generate the analog signals IN, representing components of the last calculated interference vector from the digital string DIN output by the microprocessor CPLD.

[0055] The inverting circuit INVERT of FIG. 8 is depicted in detail in FIG. 10. Substantially, it is composed of an array of inverters HB3, ..., HB10 which may be formed as shown in FIG. 11, that generates all the components O11, ..., O82 of a new superposition vector to be processed from the odd components IN1, ..., IN8.

[0056] Basically, the microprocessor unit performs the following functions:

1) drives correctly the converters;
2) acquires digital values and evaluates the Shannon entropy S;
3) compares S with a fixed threshold;

4) if S < threshold, it stops iterations and sends the results to a LED Matrix display, otherwise the results are sent (DIN) to a digital/analog converter; and

5) provides for an initial condition of superposed basis vectors.

[0067] A display may also be connected to the CPLD for displaying results. If the Shannon entropy is not sufficiently small, the binary string has to be re-converted in an analog signal by a digital/analog converter to feed it back into the entanglement subsystem I-a. In the embodiment of FIG. 8, the digital/analog converter was the commercial device AD7228 of Analog Devices.